FASTC-ID(RS)T-0310-92

FOREIGN AEROSPACE SCIENCE AND TECHNOLOGY CENTER

GaAs COMPUTER TECHNOLOGY (1)

by

Wang Qiao-yu

HUMAN TRANSLATION

FASTC-ID(RS)T-0310-92 7 January 1993

GaAs COMPUTER TECHNOLOGY (1)

By: Wang Qiao-yu

English pages: 6

Source: Weidianzi Xueyu Jisuanji, Vol. 8, Nr. 6, 1991; pp. 43-47

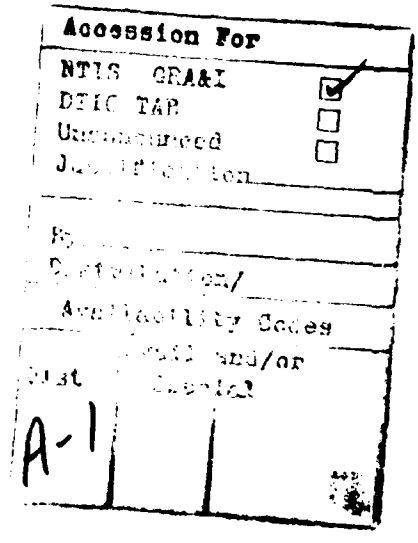
Country of origin: China
Translated by: SCITRAN
F33657-84-D-0165
Requester: FASTC/TATC/Richard H. Clark

PREPARED BY:

TRANSLATION DIVISION
FOREIGN AEROSPACE SCIENCE AND TECHNOLOGY CENTER
WPAFB, OHIO

GRAPHICS DISCLAIMER

All figures, graphics, tables, equations, etc. merged into this translation were extracted from the best quality copy available.



GaAs Computer Technology (1)

Wang Qiao-yu
(Li-Shan Microelectronics Institute)

Abstract: The paper presents development on GaAs computer and approach to designing GaAs computer.

Key Words: GaAs, Computer, Technology

/44

2. Technological foundation and computational strategy of GaAs computer

2.1 The effect of the properties of GaAs computers on the structure design of GaAs computers.

2.1.1 Main properties of GaAs E/D MESFET.

(1) Fast gate switching rate. This is because the high electron migration rate of GaAs material. The electron migration rate of GaAs materials is 6-8 times of that of silicon material.

(2) Low finished product rate. This is because of the large defect density, brittleness, processing difficulty, and high energy dissipation of GaAs material. This limits the number of transistors on a single wafer to one tenth of those of MOS wafer (about 10,000 gates). The characteristic of low number of transistors on GaAs wafers limits the ability to design GaAs wafer. Designers have to leave those functions conventionally done by the hardware to the software.

(3) The unique high speed advantage of GaAs units inside a wafer is not matched by the speed outside the wafer. The ratio between the storing-retrieving delay of the memory of GaAs units outside the wafer and the data channel delay inside the wafer is far greater than that of silicon units. Hence it is highly demanding on the compiler programming of GaAs computers.

/45

(4) Low gate input and output. This severely affects the high speed buffer memory design of the executing unit.

2.1.2. Structure selection of GaAs computers.

It is not suitable to use multi-wafer architecture in GaAs computers. This is because too much inter-wafer communication will greatly reduce or annihilate the inherent high speed advantage of the wafer. Use of single wafer processor structure can reduce the inter-wafer communication to the minimum so that the high speed potential can be utilized to the maximum.

2.1.3 Design path of single wafer GaAs processors.

Due to the limitation of number of transistors, the most suitable structure for single wafer processors is the RISK structure. The number of transistors required is much less than that of the CISC structure.

The main characteristics of RISC structure are: concise command system, fewer commands and with fixed length, single cycle execution, use of register-register operating mode, and data storage through input and storing command.

Because of the simplicity of the command system, RISC structure uses hardware control rather than micro-code method. Hence the control logic (such as UCB) only takes up 10% of the wafer area compared with 68% of the total area taken by the control logic in a CISC structure (such as MC6800). The RISC design tries to transfer the work load as much as possible from execution time to compilation time. The emphasis is on optimization problem of the compilation program. With the optimum compilation program, RISC structure is the best structure for GaAs processors. It can execute high level language with extremely fast speed.

2.1.4 Optimization of the flow line design

Whether the flow line design of GaAs processors reaches optimum directly influences the design of execution units, register files, command system, and exterior storing system. In silicon processors, the assembly line production method is achieved through increasing the data channel utilization rate of processors. In a GaAs processors, however, it is difficult to make use of data channel because of the very high ratio between the storing-retrieving time of command storing unit and the transmission delay time of the data channel. Fig. 1 shows the GaAs implementation of UCB-RISK production line.

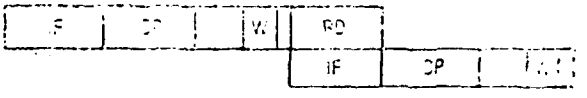


Fig. 1 GaAs implementation of UCB-RISC flow line.

In GaAs processors, the command retrieving delay is much larger than the data channel delay. If the flow line method of silicon processors is directly copied to GaAs processors, it will cause the phenomenon of severe reduction in data channel utilization rate. This is because that the switching speed in a GaAs wafer is nearly ten times of that in a silicon wafer. In the mean time, the storing-retrieving delay of the storing unit outside the wafer sees only a tiny improvement over that of a silicon wafer. The key in the design of GaAs computer structure is maximum utilization of the high switching rate advantage in a GaAs wafer.

The optimization problem in the design of GaAs production line is to find a way to secure that the command bandwidth can meet the need of the data channel. For that reason, the following two methods are used:

(1) Use following methods to increase the command information bandwidth: (A) Set the command storing unit at a high layer than in a silicon wafer. Given the number of layers of the storing unit, the storing-retrieving time difference between different layers in a GaAs processor is

large than that in a silicon processor. Through increasing the layers, the average storing-retrieving time of the storing unit is greatly reduced. (B) Use flow line storing unit systems. (C) Use inserting storing unit system. Rewrite multiple words to faster and smaller storing unit from slower and larger storing unit, or directly rewrite to the processor. This technique, for example, is used is IBM 3033 silicon computers. (D) Use multiple command technique. This technique is used in SU-M1PS silicon computers. There is a potential value in using this technique in GaAs computers. Multi-retrieving commands require more terminals in packaging. They also require more concise command format.

(2). To increase data channel utilization time, cut some sources (transistor, wafer area) from the data channel functional sources and use it somewhere else. For each retrieving command, make repeated use of the execution unit.

2.1.5 Register file design optimization

In GaAs processors, use of large register files on the wafer has an extremely important effect on improving performance. Because of the very large ratio between memory storing-retrieving delay outside the wafer and the register storing-retrieving delay, large register files can greatly reduce storing-retrieving delay during operation. Reducing command size is even more important to GaAs processors. Concise programs increase the effectiveness of command cache and other parts. GaAs processors have pour-in/store structure. Hence the number of times executing pour-in/store command are reduced. Furthermore, there is no need to store into command memory.

In GaAs processors, storing-retrieving delay outside the wafer may adversely damage the performance, therefore the design of good quality large register files is am very important means of obtaining high performance.

/46

A simple register unit of GaAs register files using only one read main line is shown in Fig. 2.

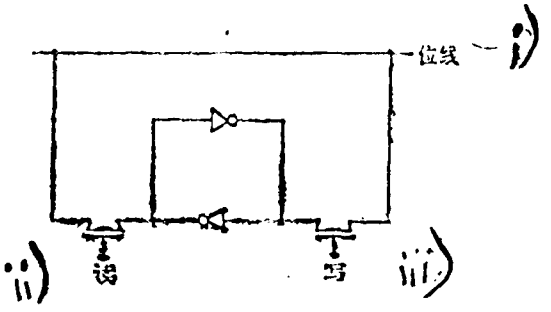


Fig. 2 Single read line register unit. i) position line, ii) read, iii) write.

In designing GaAs processors, it is important to select appropriate register file structure. For GaAs processors, single window register file structure has inherent limitations in reducing file cut and change expenditure and in reducing register file usage. Multi-window register file structure

is very attractive in reducing communication outside the wafer, but it needs large number of registers, which is too costly to be used in GaAs computers.

Recently developed multi-window register file structure with variable window size (register file usage is 100%) and multi-window register file structure with background input/store both have distinctive advantages. When designing register files, one should adopt the register file structures most appropriate for the use of GaAs computers based on application environment and manufacturing technology.

Using capacity-difference technique in designing register files can greatly improve the product rate of GaAs processors.

Single window structure generally uses compiler technology to achieve capacity-difference. The way to do this is that the compiler program sets variables only in good register units. Its disadvantages are: (1) small register file causes performance to deteriorate; (2) compiler program of one machine cannot be used in another machine.

Multi-window file structure solves capacity-difference problem through increasing small running time expenditure. It needs a self monitoring program to self monitor, to see wether there exists bad register.

2.1.6 Design optimization of execution units

Unlike silicon processors, the execution unit design of GaAs processors cannot use transistor resources to speed up primitive operations. GaAs wafers lack area resources. Communications outside the wafer is the bottleneck of its performance. Hence there is no benefit in speeding up primitive operation speed.

Use two ways to design execution unit of GaAs processors. One is reducing resources assigned to execution unit (no matter whether or not it reduces the execution speed of primitive operations). In fact, it won't damage execution speed. As long as the communication outside the wafer plays a secondary role, the resources saved can be used else where. Another way is increasing resources used in execution but limited to complicated operations only to enhance performance. Once the execution of complicated operations is sped up, the command redundance in executing complicated operations is reduced.

GaAs adder design should use impulse-carry and carry-selection. Pre-carry method is not suitable for GaAs adders.

GaAs multiplier and divider can effectively use simple hardware add/subtract and shifting method as well as "Boss" multiplication method. It is best for the shifter design to use barrel shaped shifter.

2.1.7 Design optimization of command system

The command system design of GaAs microprocessors requires: command system simplicity, fewer command format types, same command field used by each command format,

operation itself preserving primitive operation. However, command system simplification is in conflict with program complexification.

GaAs processor wafer has small number of transistors, therefore the command decoding and controlling circuit have to be made as small as possible. This is the reason why RISC technology is used. To guarantee single cycle execution of GaAs RISC structure, all complicated operations have to be processed together starting from primitive operations.

The long propagation delay outside GaAs processor wafers is the cause of the benefits of small programs. Small command has large influence on performances. To reduce command size, three techniques are needed: quasi-Huffman variable length coding technique, substitution of common command sequences with single command, and selection of optimized digital address seeking mechanism.

2.2 Design technology analysis of high performance GaAs signal processors

If the VLSI design technology of silicon is directly copied to the functional parts of GaAs, the high speed strength of GaAs will not be utilized. In designing GaAs signal processors, it is necessary to consider the high speed characteristics of GaAs gates and high clock frequency work condition. Structural consideration is even more important especially in using second generation GaAs transistors to manufacture gate circuits with propagation delay between 10-50 ps.

/47

2.2.1 Overall design method

For GaAs processors, because of the use of high clock frequency, one must simultaneously consider, during the design, problems in the fields of IC packaging, system cooling, communication inside the processors, power dissipation, and others. To achieve high performance signal processors, one must follow overall design method.

The overall design method requires that the system design enable the data to flow uniformly and continuously between neighboring IC block and to flow between neighboring logic boards so that the resulting data path length is the shortest and the throughput is the largest. This design layout leads to logic boards and wafer packaging with the least complexity, and the lowest cost. On multi-layer printed circuit boards, only two or three layers of signal lines are needed instead of ten layers. This has very important implications in cost control and performance enhancement. Almost all of the interconnections between GaAs devices are in the form of transmission lines. All of the terminals of transmitting lines use transistor network, which dissipates fairly large amount of power. Therefore one should minimize the interconnection lines between logic board during the design and make sure that all connecting lines are working at the largest bit-rate (1-2 mil bit/sec/line)

2.2.2 Signal processing algorithm

At present, a signal processing algorithm for fixed precision operation data flow has resulted from research. Operation data flow executes all data processing functions through module bit impulsive contraction array. Scalor multiplication, array-matrix multiplication, matrix-matrix multiplication, linear transformation, rotation and affine transformation, extraction of characteristic values and characteristic vectors can be done with bit decomposition method. This bit decomposition algorithm is especially suited for GaAs signal processors.

References

[1] Clifrd Barney, Electronics Week. p.36" 1985.4.

[2] Ron Wilson, Computer Design. p.30. 1989.1.

[3] Eric R. Fox et al.. Computer. p. 71. 1986, 10.

[4] Weljko Milutinovic et al., Computer. p.45, 1986.10.

[5] Barry K. Gilbert et al., Computers, p.29, 1986.10.

[6] Weljko Milutinovic et al., Computer, p.30, 1986.3.

(compiled by Hu Junpeng)

DISTRIBUTION LIST

DISTRIBUTION DIRECT TO RECIPIENT

ORGANIZATION	MICROFICHE
B085 DIA/RTS-2FI	1
C509 BALLOC509 BALLISTIC RES LAB	1
C510 R&T LABS/AVEADCOM	1
C513 ARRADCOM	1
C535 AVRADCOM/TSARCOM	1
C539 TRASANA	1
Q592 FSTC	4
Q619 MSIC REDSTONE	1
Q008 NTIC	1
Q043 AFMIC-IS	1
E051 HQ USAF/INET	1
E404 AEDC/DOF	1
E408 AFWL	1
E410 ASDIC/IN	1
E411 ASD/FTD/TTIA	1
E429 SD/IND	1
P005 DOE/ISA/DDI	1
P050 CIA/OCR/ADD/SD	2
1051 AFIT/LDE	1
CCV	1
P090 NSA/CDB	1
2206 FSL	1

Microfiche Nbr: FTD93C000039L
FTD-ID(RS)T-0310-92